HomeWork
106034061
曾靖渝

Exercise #2. we solve this problem by my program

```
Weights:-0.4 0.3 0.1
Program ended with exit code: 010
Example:110
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.3 0.1
11
Example:111
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.2 0.5 0.3
00
Example:100
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.5 0.3
->Modify:1 times
```

as we can see. We find perfect solution by only modify formula one times

Computer Assignment #2.

Eta=0.2:

```
#include <stdio.h>
#include <iostream>
#include <string>
#include <vector>
#include <set>
#define ETA 0.2
```

```
c(x)=1
Weights:-0.2 0.2 0 0.4 0 0.2
Example:101000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2
Example:100000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2
Example:101010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2
Example:100110
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0.2 0 0.2 -0.2 0.2
Example:111000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0.2 -0.2 0.2
Example:100010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0.2 -0.2 0.2
Example:101011
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.2 0.2 0.2 0.2 0 0.4
Example:100011
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0.2 0.2 0.2 -0.2 0.2
->Modify:7 times
```

Eta=0.4:

```cpp
#include <stdio.h>
#include <iostream>
#include <string>
#include <vector>
#include <set>
#define ETA 0.4
```

```
c(x)=1
Weights:-0.2 0.2 0.2 0.6 -0.2 0.6
Example:101000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0.2 0.6 -0.2 0.6
Example:100000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0.2 0.6 -0.2 0.6
Example:101010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0.2 0.6 -0.2 0.6
Example:100110
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.6 0.2 0.2 0.2 -0.6 0.6
Example:111000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0.2 0.2 0.2 -0.6 0.6
Example:100010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0.2 0.2 0.2 -0.6 0.6
Example:101011
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.2 0.2 0.6 0.2 -0.2 1
Example:100011
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.6 0.2 0.6 0.2 -0.6 0.6
->Modify:6 times
```

Eta=0.6:

```cpp
#include <stdio.h>
#include <iostream>
#include <string>
#include <vector>
#include <set>
#define ETA 0.6
```

```
c(x)=1
Weights:-0.4 0.2 0.2 0.8 -0.4 0.8
Example:101000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0.2 0.8 -0.4 0.8
Example:100000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0.2 0.8 -0.4 0.8
Example:101010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0.2 0.8 -0.4 0.8
Example:100110
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-1 0.2 0.2 0.2 -1 0.8
Example:111000
->Classifying..........
h(x)=0
c(x)=0
Weights:-1 0.2 0.2 0.2 -1 0.8
Example:100010
->Classifying..........
h(x)=0
c(x)=0
Weights:-1 0.2 0.2 0.2 -1 0.8
Example:101011
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.4 0.2 0.8 0.2 -0.4 1.4
Example:100011
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-1 0.2 0.8 0.2 -1 0.8
->Modify:6 times
```

Eta=0.8:

```
#include <stdio.h>
#include <iostream>
#include <string>
#include <vector>
#include <set>
#define ETA 0.8
```

```
c(x)=1
Weights:-0.6 1 0.2 1 0.2 0.2
Example:101000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 1 0.2 1 0.2 0.2
Example:100000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 1 0.2 1 0.2 0.2
Example:101010
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 1 0.2 1 0.2 0.2
Example:100110
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-1.4 1 0.2 0.2 -0.6 0.2
Example:111000
->Classifying..........
h(x)=0
c(x)=0
Weights:-1.4 1 0.2 0.2 -0.6 0.2
Example:100010
->Classifying..........
h(x)=0
c(x)=0
Weights:-1.4 1 0.2 0.2 -0.6 0.2
Example:101011
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.6 1 1 0.2 0.2 1
Example:100011
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-1.4 1 1 0.2 -0.6 0.2
->Modify:8 times
```

According to the outcome of statics, we can speculate that if the Eta is larger, the influence of each modification is large. As a result, we can assume if the Eta is too large, we should modify the formula more times.

Computer Assignment #3.

N=1:

```
int main(void)
{
    int attribute_number=6;
//    cin>>attribute_number;
    int instance_number=20;
//    cin>>instance_number;
    perception_learning Classifier(attribute_number, instance_number);
//    int i=0;
```

```
110011
000001
111111
110101
100100
010010
101010
011100
110111
011101
100111
111100
010001
000001
010100
001100
110000
000101
010110
000110
```

```
c(x)=1
Weights:-0.2 0.2 0 0.4 0 0.2 0
Example:1010001
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2 0
Example:1000001
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2 0
Example:1010100
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.4 0 0.2 0
Example:1001100
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0.2 0 0.2 -0.2 0.2 0
Example:1110000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0.2 -0.2 0.2 0
Example:1000101
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0.2 -0.2 0.2 0
Example:1010110
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.2 0.2 0.2 0.2 0 0.4 0
Example:1000110
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0.2 0.2 0.2 -0.2 0.2 0
->Modify:7 times
```

N=5:

Compare N=1 and N=5 we can see the length of attribute also affect the modifying times. According the textbook, if the length attributes is n times, the modifying also will be n times.

```
//N=5
/*
 1100110101
 0000010100
 1111111100
 1101011001
 1001000000
 0100100000
 1010111111
 0111011100
 1101111011
 0111000111
 1001100011
 1111001010
 0100000010
 0000011000
 0101000000
 0011000000
 1100000000
 0001000001
 0101101010
 0001100000
```

```
Example:10100000010
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0 0 0.4 0.2 0 0.2 0.4 0.2 0 0.2
Example:10000011000
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.6 0 0 0.4 0.2 0 0 0.2 0.2 0 0.2
Example:10101000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0 0 0.4 0.2 0 0 0.2 0.2 0 0.2
Example:10011000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0 0 0.4 0.2 0 0 0.2 0.2 0 0.2
Example:11100000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0 0 0.4 0.2 0 0 0.2 0.2 0 0.2
Example:10001000001
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.6 0 0 0.4 0.2 0 0 0.2 0.2 0 0.2
Example:10101101010
->Classifying..........
h(x)=0
c(x)=1
->Modify Weight
Weights:-0.4 0 0.2 0.4 0.4 0.2 0 0.4 0.2 0.2 0.2

Example:10001100000
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.6 0 0.2 0.4 0.2 0 0 0.4 0.2 0.2 0.2
->Modify:8 times
```

N=10:

```
//N=10
//1111111111
//0000000000
/*
 110011111111111
 000000000000000
 111111111111111
 110101111111111
 100100000000000
 010010000000000
 101011111111111
 011101111111111
 110111111111111
 011101111111111
 100110000000000
 111101111111111
 010000000000000
 000000000000000
 010100000000000
 001100000000000
 110000000000000
 000100000000000
 010111111111111
 000110000000000
```

```
h(x)=1
c(x)=1
Weights:-0.2 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1010000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1010100000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1001100000000000
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0.2 0 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1110000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1000100000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:1010111111111111
->Classifying..........
h(x)=1
c(x)=1
Weights:-0.4 0.2 0 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2

Example:1000110000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0.2 0 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
->Modify:5 times
```

N=15:

```
//N=15
//000000000000000
//111111111111111
/*
 1100111111111111111
 0000000000000000000
 1111111111111111111
 1101011111111111111
 1001000000000000000
 0100100000000000000
 1010111111111111111
 0111011111111111111
 1101111111111111111
 0111011111111111111
 1001111111111111111
 1111011111111111111
 0100000000000000000
 0000000000000000000
 0101000000000000000
 0011000000000000000
 1100000000000000000
 0001000000000000000
 0101111111111111111
 0001100000000000000
```

```
Example:11111011111111111111
->Classifying..........
h(x)=1
c(x)=1
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10100000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10101000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10011000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:11100000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10001000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10101111111111111111
->Classifying..........
h(x)=1
c(x)=1
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:10001100000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.2 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
->Modify:3 times
```

N=20:

```
//N=20
//11111111111111111111
//00000000000000000000
/*
 1100111111111111111111111
 0000000000000000000000000
 1111111111111111111111111
 1101011111111111111111111
 1001000000000000000000000
 0100100000000000000000000
 1010111111111111111111111
 0111000000000000000000000
 1101111111111111111111111
 0111011111111111111111111
 1001111111111111111111111
 1111011111111111111111111
 0100000000000000000000000
 0000000000000000000000000
 0101000000000000000000000
 0011000000000000000000000
 1100000000000000000000000
 0001000000000000000000000
 0101111111111111111111111
 0001100000000000000000000
 */
```

```
->Classifying..........
h(x)=1
c(x)=1
Weights:-0.2 0 0.2 0.4 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:101000000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0 0.2 0.4 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:100000000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.2 0 0.2 0.4 0.2 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:101010000000000000000000
->Classifying..........
h(x)=1
c(x)=0
->Modify Weight
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:100110000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:111000000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:100010000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:101011111111111111111111
->Classifying..........
h(x)=1
c(x)=1
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
Example:100011000000000000000000
->Classifying..........
h(x)=0
c(x)=0
Weights:-0.4 0 0 0.4 0 0 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
->Modify:5 times
```

According the textbook, if the length attributes is n times, the modifying also will be n times. However, the statics doesn't implied it. We can blame it on our own attributes example. If we randomly make the example, the outcome may be more confident.

How I design My Program:

First, I declaim a class to do perception learning, and us vectors to store example

```
using namespace std;
using example=vector<int>;
class perception_learning//Perception Learning Classifier(Linear Seperabitlity)
```

```
set<example> Training_Set;
vector<int> class_labels;
vector<double> weight;//Memorize all of Weights
int c_x=0;//c(x)
int h_x=0;//h(x)
int attribute_number;
int instance_number;
int modifying_number;//Memorize Modifying times
```

Here is the private variable in the class.

```
int attribute_number=5;
  cin>>attribute_number;
int instance_number=20;
  cin>>instance_number;
perception_learning Classifier(attribute_number, instance_number);
  int i=0;
```

Called the class to Classify in main function.

```
int compute_ClassLabel(example ex)//Compute c(x)
{
    int count=0;
    for(auto i:ex)
        if(i==1)count++;
    if(count>3)c_x=1;
    else c_x=0;
    cout<<"c(x)="<<c_x<<endl;
    return c_x;
}
int compute_H_ClassLabel(example ex)//Compute h(x)
{
    double sum=0;

    for(int i=0;i<=attribute_number;i++)
    {
        sum+=ex[i]*weight[i];
    }
    if(sum>0)h_x=1;
    else h_x=0;
    cout<<"h(x)="<<h_x<<endl;
    return h_x;
}
void Classifying(example ex)//Classifying
{
//        cout<<"Attribute Number:"<<attribute_number<<" ,Instance Number:"<<instance_number<<endl;
    cout<<"->Classifying..........."<<endl;
    compute_H_ClassLabel(ex);
    compute_ClassLabel(ex);
    if(h_x!=c_x)modify_weight(ex);
    print_weight();
}
void modify_weight(example ex)//Refresh Weights
{
    cout<<"->Modify Weight"<<endl;
    for(int i=0;i<=attribute_number;i++)
    {
        weight[i]+=ETA*(double)ex[i]*(double)(c_x-h_x);
    }
    modifying_number++;
}
```

I use four main function to complete the work. The introduction is in the picture.